AMENDMENTS TO THE CLAIMS

1. (Previously Presented) A method for processing signal values in a digital signal processor, the method comprising:

in response to a single instruction that specifies a plurality of signal values and a plurality of code segments of a despreading code:

complex multiplication of each signal value by a respective one of the code segments to provide a plurality of intermediate results;

complex addition of the intermediate results to provide a despread result; and

storing the despread result, wherein the complex multiplication, the complex addition and the storing of the despread result are executed in a single clock cycle of the digital signal processor.

2. (Previously Presented) A method as defined in claim 1 further comprising, in response to the single instruction, including a previous result from a previous instruction in the complex addition which provides the despread result.

3. (Previously Presented) A method as defined in claim 1, wherein the despreading code has a spreading factor divisible by 4.

4. (Previously Presented) A method as defined in claim 1, wherein the despreading code is divided into code segments, each code segment having comprising a 2 bit complex code comprising 1 real bit and 1 imaginary bit.

5. (Previously Presented)) A method as defined in claim 4, wherein a set code bit represents a value of –1 and a clear code bit represents a value of +1.

6. (Previously Presented) A method as defined in claim 1, wherein the signal value comprises 16 bits.

7. (Previously Presented) A method as defined in claim 6, wherein the signal value comprises 8 real bits and 8 imaginary bits.

8. (Previously Presented) A method for calculating a data set in a digital signal processor, the method comprising the steps of:

in response to a single instruction that specifies a plurality of signal values and a plurality of code segments:

complex multiplication of each signal value by a respective one of the code segments to provide a plurality of intermediate results;

complex addition of the intermediate results to provide an instruction result; and

storing the instruction result, wherein the complex multiplication, the complex addition and the storing of the instruction result are executed in a single clock cycle of the digital signal processor.

9. (Previously Presented) A method as defined in claim 8, wherein, in response to the single instruction, the complex addition further comprises adding a previous result from a previous instruction to the intermediate results to provide the instruction result.

10. (Previously Presented) A method as defined in claim 8, wherein the set of codes has a spreading factor divisible by 4.

11. (Previously Presented) A method as defined in claim 8, wherein each one of the set of codes is a 2 bit complex code comprising 1 real bit and 1 imaginary bit.

12. (Previously Presented) A method as defined in claim 11, wherein a set code bit represents a value of –1 and a clear code bit represents a value of +1.

13. (Previously Presented) A method as defined in claim 8, wherein the signal value comprises 16 bits.

14. (Previously Presented) A method as defined in claim 13, wherein the signal value comprises 8 real bits and 8 imaginary bits.

15. (Cancelled).

16. (Previously Presented) A digital signal processor comprising:

a memory for storing instructions and operands for digital signal computations;

a program sequencer for generating instruction addresses for fetching selected ones of said instructions from said memory;

a computation block comprising a register file for temporary storage of operands and results and an execution block for executing a single decoding instruction that specifies a plurality of signal values and a plurality of code segments, said execution block comprising a complex multiply and accumulate engine for multiplying portions of the data signal by the code and accumulating the results; and

wherein, in response to an execution of the single decoding instruction the digital signal processor decodes the signal values by:

performing a complex multiplication of each signal value by a respective one of the code segments to provide a plurality of intermediate results,

performing an a complex addition of the intermediate results to provide a despread result, and

storing the despread result, wherein the complex multiplication, the complex addition and the storing of the despread result are executed in a single clock cycle of the digital signal processor.

17. (Previously Presented) A digital signal processor as defined in claim 16, wherein the code segments have a spreading factor divisible by 4.

18. (Previously Presented) A digital signal processor as defined in claim 16, wherein each code segment comprises a 2 bit complex code comprising 1 real bit and 1 imaginary bit.

19. (Previously Presented) A digital signal processor as defined in claim 18, wherein a set code bit represents a value of –1 and a clear code bit represents a value of +1.

20. (Previously Presented) A digital signal processor as defined in claim 16, wherein the plurality of signal values comprise 16 bits.

21. (Previously Presented) A digital signal processor as defined in claim 16, wherein the plurality of signal values comprise 8 real bits and 8 imaginary bits.

22. (Previously Presented) A method for calculating output data in a digital signal processor, the method comprising the steps of:

in response to a single instruction that specifies at least a set of complex first operands each one of the first operands comprising 8 real bits and 8 imaginary bits and a set of complex second operands each one of the second operands comprising 1 real bit and 1 imaginary bit:

performing a complex multiplication of each one of the first operands by a respective one of the second operands to provide a plurality of intermediate results;

performing a complex addition of the intermediate results to provide an instruction result; and

outputting the instruction result, wherein the complex multiplication, the complex addition and the outputting the instruction result are executed in a single clock cycle of the digital signal processor.

23. (Previously Presented) A method as defined in claim 22, wherein a set bit in one of the second operands represents a value of –1 and a clear bit in one of the second operands represents a value of +1.

24. (Previously Presented) A method as defined in claim 22, wherein the set of complex second operands comprises a despreading code.

25. (Previously Presented) A method as defined in claim 22, wherein the set of complex first operands comprises an incoming data signal.

26. (Previously Presented) A method as defined in claim 25, wherein the incoming data signal is a voice transmission signal.

27. (Previously Presented) A method for processing a signal value in a digital signal processor, comprising the step of:

in response to a single instruction executed within a single clock cycle of the digital signal processor, specifying a complex signal value and a two bit complex code segment, performing a complex multiply of the signal value by the code segment to provide a processed data value.

28. (Previously Presented) A method for processing signal values in a digital signal processor comprising the steps of:

(a) in response to a single instruction executed within a single clock cycle of the digital signal processor, specifying a set of complex signal values and a corresponding set of complex code segments, performing a complex multiply of each signal value by a corresponding code segment to provide a set of intermediate values; and

(b) in further response to said single instruction performing complex addition of the intermediate values to provide a processed signal value.

29. (Previously Presented) A method as defined in claim 28 further comprising the steps of repeating steps (a) and (b) for a plurality of sets of complex signal values to provide a stream of processed signal values.

30. (Previously Presented) A method as defined in claim 28 further wherein each of the complex code segments is a two bit complex code.